

Claims

1. (Currently Amended) A computer program product embodied on a first computer-readable medium and comprising code that, when executed, causes a computer to perform a method of generating a partial procedure summary of a procedure of multithreaded software, wherein the procedure performs a plurality of actions when executed, the method comprising:

during a reachability analysis of at least a portion of the multithreaded software, reaching the procedure at a first point in the reachability analysis;

when the procedure is reached at the first point in during the reachability analysis,

1) identifying at least one plurality of the actions within the procedure as atomically modelable with respect to multithreaded execution of the procedure as atomically modelable actions, wherein the atomically modelable actions are not subject to interruption by other threads, and

2) generating, by the computer, the at least one partial procedure summary of the procedure from the identified atomically modelable actions, wherein the at least one partial procedure summary comprises at least one state pair, wherein the at least one state pair models an initial state and a resulting state of the identified atomically modelable actions;

storing the at least one partial procedure summary of the procedure of multithreaded software on a second computer readable medium;

during continuation of the reachability analysis, at a second point in the reachability analysis, reaching the procedure a second time; and

wherein if an initial state of the procedure reached the second time is the same as the initial state of the at least one partial procedure summary, then the resulting state of the at least one partial procedure summary is used as a state of the procedure state in the reachability analysis.

2. (Canceled)

3. (Previously Presented) The computer program product of claim 1, the method further comprising:

modeling execution of the software such that the state pair that comprises the partial procedure summary is executed rather than the atomically modelable actions.

4. (Previously Presented) The computer program product of claim 1, the method further comprising:

during modeling, comparing an indicated state invariant with a modeled state; and responsive to determining the modeled state fails the indicated state invariant, indicating that a programming flaw is present in the software; wherein determining the modeled state fails the indicated state invariant comprises determining that a condition is false for at least one execution path.

5. (Previously Presented) The computer program product of claim 1, wherein the resulting state comprises at least one of a plurality of possible states of the multithreaded software after execution of the modeled procedure, the method further comprising:

storing an initial program counter location within the modeled procedure for the initial state;

storing a resulting program counter location within the modeled procedure for the resulting state; and

associating the initial program counter location within the modeled procedure and the resulting program counter location within the modeled procedure with the partial procedure summary.

6. (Previously Presented) The computer program product of claim 1 wherein the reachability analysis consulting the at least one partial procedure summary further comprises determining possible execution paths within the procedure and using the procedure summary to explore possible states.

7. (Previously Presented) The computer program product of claim 1 wherein the identifying comprises identifying a transaction boundary within the actions.

8. (Previously Presented) The computer program product of claim 1 wherein the identifying comprises identifying at least one of the plurality of actions as movable later in time with respect to actions executed by other threads without affecting a resulting end state.

9. (Previously Presented) The computer program product of claim 1 wherein the identifying comprises identifying a sequence of actions having zero or more right movers followed by an atomic action followed by zero or more left movers.

10. (Previously Presented) The computer program product of claim 1 wherein the plurality of actions atomically modelable with respect to multithreaded execution of the software is a subset of the plurality of actions of the procedure, the subset comprising less than all of the plurality of actions of the procedure.

11. (Currently Amended) A computer program product embodied on a first computer-readable medium and comprising code that, when executed, causes a computer to perform a method of modeling multithreaded software, the method comprising:

performing a reachability analysis of the multithreaded software;

during the reachability analysis, reaching a procedure at a first point in the reachability analysis;

analyzing actions of the multithreaded software within the procedure such that actions that can be executable atomically within the procedure are determined, wherein actions that can be executable atomically are not subject to interruption by other threads;

based on the analyzing, the computer generating a plurality of partial procedure summaries for the multithreaded software, the plurality of partial procedure summaries comprising respective start and end actions for the determined actions executable atomically; and

during the reachability analysis, ~~again~~ reaching the procedure at a second point in the reachability analysis and reusing the plurality of partial procedure summaries to determine actions executable atomically;

storing in a second computer-readable medium at least some of the plurality of generated partial procedure summaries for the multithreaded software;

wherein the partial procedure summaries comprise ~~comprises~~ a plurality of modeled states of the multithreaded software for multithreaded execution of the multithreaded software.

12. (Previously Presented) The computer program product of claim 11 wherein at least one of the partial procedure summaries comprises at least two or more partial procedure summaries summarizing a procedure.

13. (Previously Presented) The computer program product of claim 11 wherein at least one of the partial procedure summaries comprises at least one partial procedure summary for a procedure, wherein the partial procedure summary summarizes less than all of the procedure.

14. (Previously Presented) The computer program product of claim 11 wherein the analyzing comprises:

identifying a series of transactions within the multithreaded software; and
modeling the transactions via the partial procedure summaries.

15. (Currently Amended) A ~~computer-implemented~~ system for modeling multithreaded software, the system comprising:

a microprocessor;

a first computer-readable medium containing instructions configured to reconfigure the microprocessor to act as at least a portion of a model checker operable to analyze a model of the multithreaded software via checking the model of the multithreaded software for programming flaws, the model checker comprising:

the model of the multithreaded software, wherein the model comprises[[:]]
a plurality of partial procedure summaries modeling beginning states and ending states of partial summaries of procedures within the multithreaded software during multithreaded execution of the multithreaded software, the partial procedure

summaries comprising the start and end states of sets of actions within the procedures, the actions atomically modelable with respect to multithreaded execution of the software in that the atomically modelable actions will all be performed within a single procedure by a same thread; and

a reachability analyzer operable to employ upon reaching the sets of actions at least a second time during a reachability analysis, one or more of the partial procedure summaries instead of the sets of actions to generate modeled states of the software; and

a second computer-readable medium containing instructions configured to cause the microprocessor to store in a third computer-readable medium an indication of one or more programming flaws identified by the model checker.

16. (Previously Presented) The computer-implemented system of claim 15 wherein at least one of the partial procedure summaries comprises a partial procedure summary summarizing actions deemed to have occurred one after another without interruption by another thread.

17. (Canceled)

18. (Previously Presented) The computer-implemented system of claim 16 wherein the system is operable to detect programming flaws via comparing an indicated state invariant with the modeled states.

19. (Canceled)

20. (Canceled)

21. (Previously Presented) The computer program product of claim 1, wherein the at least one state pair comprises an initial state of the procedure and at least one of a plurality of possible states of the multithreaded software after execution of the modeled procedure.

22. (Canceled)

23. (Previously Presented) The computer program product of claim 1, wherein the resulting state is in a location in the procedure prior to the location of the initial state in the procedure.

24. (Canceled)

25. (New) A computer program product embodied on a first computer-readable medium and comprising code that, when executed, causes a computer to perform a method of generating a partial procedure summary of a procedure of multithreaded software, wherein the procedure performs a plurality of actions when executed, the method comprising:

during a reachability analysis of at least a portion of the multithreaded software, reaching the procedure at a first point in the reachability analysis;

when the procedure is reached at the first point in the reachability analysis,

identifying at least one plurality of the actions within the procedure as atomically modelable with respect to multithreaded execution of the procedure as atomically modelable actions, wherein the atomically modelable actions are not subject to interruption by other threads, wherein the identifying comprises identifying a sequence of the actions within the procedure having one or more right movers followed by an atomic action followed by one or more left movers, the identifying further comprising identifying a transaction boundary within the plurality of actions using a phase variable, the phase variable indicating whether a thread is modeled as executing at least one of the one or more left movers of the identified sequence of actions or at least one of the one or more right movers of the identified sequence of actions, and

generating, by the computer, the at least one partial procedure summary of the procedure from the identified atomically modelable actions, wherein the at

least one partial procedure summary comprises at least one state pair and the phase variable, wherein the at least one state pair models an initial state and a resulting state of the identified atomically modelable actions;

storing the at least one partial procedure summary of the procedure of multithreaded software on a second computer readable medium;

during continuation of the reachability analysis, at a second point in the reachability analysis, reaching the procedure a second time; and

wherein if an initial state of the procedure reached the second time is the same as the initial state of the at least one partial procedure summary, then the resulting state is used as a procedure state in the reachability analysis.